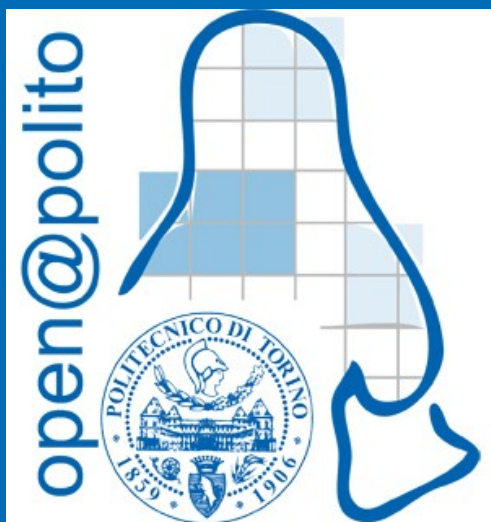




Con il supporto di:



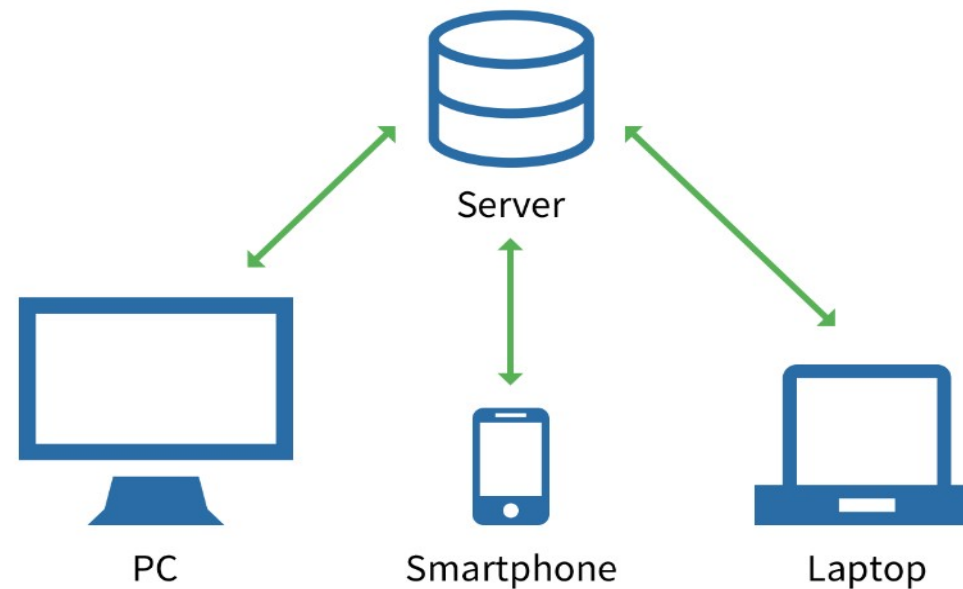
# SELF HOSTING

Di Enrico Franciscono

---

# Cos'è un server?

Un server è un computer acceso H24/7 pronto a rispondere a richieste per fornire un servizio.



# Cos'è un server?



# Cos'è un server?



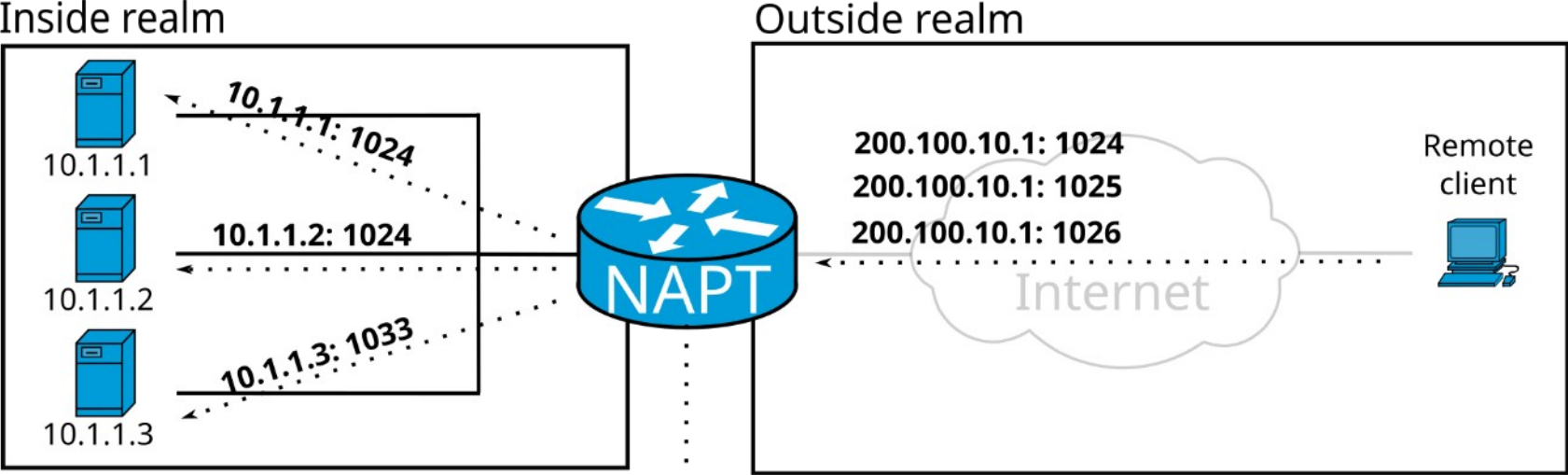
# Obiettivo

- Installare un interfaccia di configurazione
- Installare un server HTTP per hostare un semplice sito statico
- Abilitare l'HTTPS per un connessione sicura
- Permettere di configurare il server da remoto
- Aprire un semplice blog con docker

# Stato iniziale

- Installazione “pulita” di Debian
- Server SSH (abilitato durante l’installazione)
- IP pubblico statico al router di casa
- Abilitato il port-forwarding dal router al server
- Procurato dominio e configurato il DNS

# Port-forwarding



Inside realm	Outside realm
10.1.1.1: 1024	200.100.10.1: 1024
10.1.1.2: 1024	200.100.10.1: 1025
10.1.1.3: 1033	200.100.10.1: 1026

NAPT table

Michel Bakni, CC BY-SA 4.0,  
via Wikimedia Commons

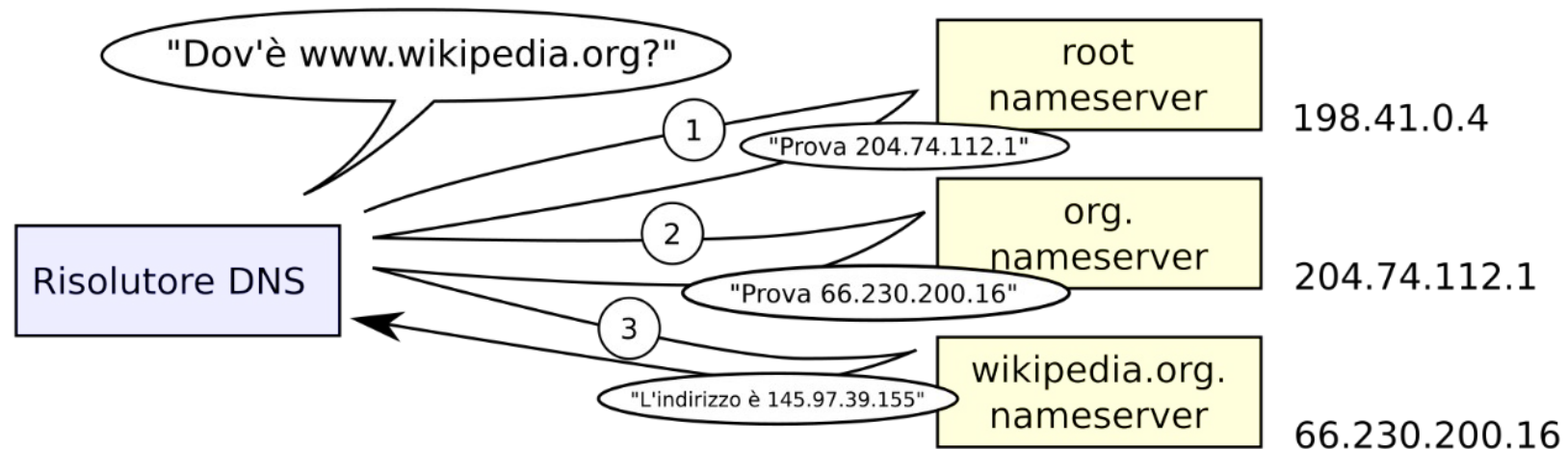
Porta 80 e 443 rediretta all'ip interno del server

# DNS

L'indice del web, associa ogni dominio ad un ip.

enrico.to.it → IP di casa

\*.enrico.to.it → IP di casa



Lion Kimbro (original author); Mess (derivative work), CC0, via Wikimedia Commons

# Cockpit

Cockpit è un interfaccia web per la gestione di uno o più server in modo intuitivo.

Al contrario di soluzioni alternative, non richiede configurazioni specifiche del sistema operativo per funzionare; Cockpit comunica con i programmi per-esistenti sul computer e rende più semplice l'interazione.



# COCKPIT

<https://cockpit-project.org/running.html>

<https://github.com/cockpit-project/cockpit-files>

# Nginx

Nginx è il server web più popolare al mondo grazie alle sue ottime performance ed alla sua grande flessibilità.

Lo utilizzeremo per due scopi:

- Hostare un sito staticamente
- Reverse proxy

# Sito statico

```
server {  
    listen 80 default_server;  
    server_name <dominio>;  
  
    root /var/www/html;  
  
    location / {  
        try_files $uri $uri/ =404;  
    }  
}
```

# Certbot

Certbot è uno script per ottenere e gestire certificati TLS (HTTPS).

Ha un'integrazione con nginx che automatizza il processo di richiesta ed aggiornamento.

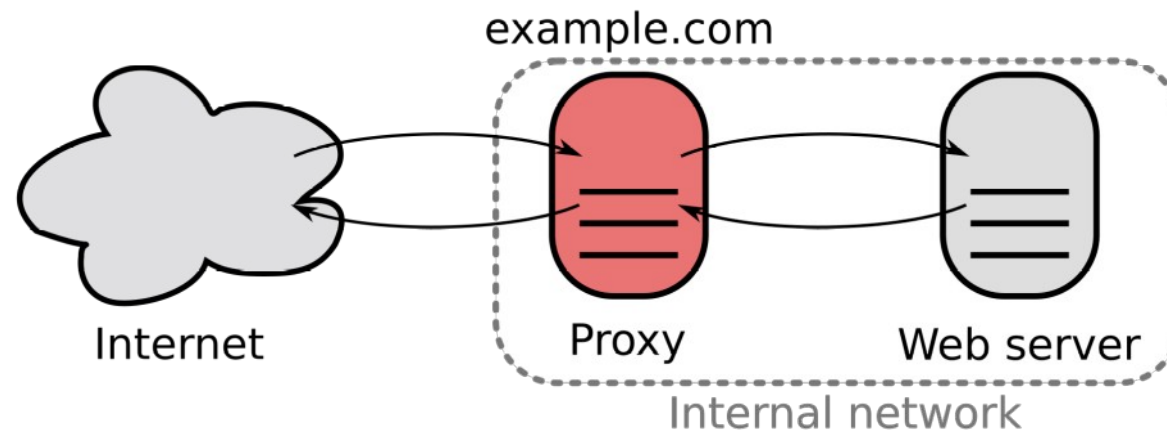
<https://certbot.eff.org>

```
https://raw.githubusercontent.com/vinylz/certbot-install/master/  
install.sh | bash
```

```
sudo certbot --nginx -d <dominio>
```

# Cockpit da remoto

Possiamo utilizzare nginx per “aggiungere” l’HTTPS a cockpit e quindi poter accedere in sicurezza da remoto all’interfaccia di configurazione del nostro server. Per far ciò, utilizzeremo un sotto dominio e ridirigeremo tutto il traffico al server cockpit. Questa configurazione viene definita “Reverse proxy”.



H2g2bob, CC0, via Wikimedia Commons

# Cockpit da remoto

```
server {  
    listen 80;  
    server_name cockpit.enrico.to.it;  
    client_max_body_size 64M;  
    location / {  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
        proxy_set_header Host $host;  
        proxy_pass http://127.0.0.1:9090;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection "upgrade";  
    }  
}
```

# Docker

Docker è una via di mezzo tra una macchina virtuale ed un programma eseguito direttamente sul sistema.

Permette di isolare i servizi, rimuovendo i conflitti dovuti a versioni diverse dei pacchetti, aumentando la sicurezza e semplificando la gestione, con un impatto ridotto alle performance.

<https://docs.docker.com/engine/install/debian/>

# Wordpress

Wordpress è la piattaforma blog più utilizzata in tutto il mondo.

È flessibile, estensibile e molto supportata.

Possiamo installarlo con docker:

[https://hub.docker.com/\\_/wordpress](https://hub.docker.com/_/wordpress)

## Altri servizi



**[matrix]**

The sky is the limit...

# Il vostro futuro

